IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

First Named Applicant: Haas Serial No.: 09/658,303 Filed: September 8, 2000 For: **SYSTEM AND METHOD FOR SCHEMA MAPPING**)))))))))) Art Unit: 2175 Examiner: Mahmoudi **ARC9-2000-0125-US1** March 12, 2004 750 B STREET, Suite 3120 San Diego, CA 92101

DECLARATION

Commissioner of Patents and Trademarks
Washington, DC 20231

Dear Sir:

I, Laura Haas, declare as follows:

1. I am a co-inventor of the invention claimed in the above-captioned U.S. patent application.

2. My co-inventors and I both conceived of and reduced the invention to practice at least as early as June 29, 2000 as evidenced by the enclosed document.

3. Specifically, the enclosed document entitled "Disclosure ARC8-2000-0176", which formed the basis for the present application and which is accurately dated as being "last modified on 05/15/2000" discloses the invention of, e.g., Claim 1 for the following detailed reasons.

4. On the second page under the second numeral "2" it is disclosed that data is mapped from a source schema into a target schema by taking as input a set of value correspondences, with each value correspondence representing a function for deriving a value of a target attribute from one or more values of source attributes. Continuing to page 3, at the top it is disclosed that value correspondences are grouped into potential sets (step #1 on page 3), and then selecting candidate sets from at least some potential sets (step #2 on page 3). Step #3 on page 3 teaches grouping at least some candidate sets into covers. Step #4 on page 3 discloses using a cover to generate a query which can be used to populate the target relation and, hence, which represents a source schema-to-target schema mapping.

5. Likewise, at least the remaining independent claims are fully disclosed in the enclosed document.

6. I further declare, based on first hand knowledge, that my co-inventors and myself were

reasonably diligent in disclosing the invention to IBM patent attorneys and promoting the filing of a patent application in accordance with standard IBM patenting procedures at least from a time prior to June 29, 2000 until that date.

7. I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United State Code and that such willful, false statements may jeopardize the validity of the application or any patent issued thereon.

Laura Haas

Respectfully submitted,

John L. Rogitz Registration No. 33,549 Attorney of Record 750 B Street, Suite 3120 San Diego, CA 92101 Telephone: (619) 338-8075

JLR:jg

33,549

Disclosure ARC8-2000-0176

Created By: Mauricio A Hernandez **Created On:** 03/16/2000 09:54:26 AM
Last Modified By: Ottie De Jesus-Dickey **Last Modified On:** 05/15/2000 01:26:51 PM

*** IBM Confidential ***

Required fields are marked with the asterisk (*) and must be filled in to complete the form .

Summary

Status	Under Evaluation
Processing Location	ARC
Functional Area	DPB - Computer Science
Attorney/Patent Professional	Marc D McSwain/Almaden/IBM
IDT Team	Marc D McSwain/Almaden/IBM; Cheryl Ruby/Almaden/IBM
Submitted Date	05/09/2000 08:04:20 PM
Owning Division Select	RES
PVT Score Calculate	To calculate a PVT score, use the 'Calculate PVT' button.
Lab	
Technology Code	

Inventors with Lotus Notes IDs

Inventors: Laura Haas/Almaden/IBM, Mauricio A Hernandez/Almaden/IBM

Inventor Name > denotes primary contact	Inventor Serial	Div/Dept	Manager Serial	Manager Name
Haas, Laura M.	043331	22/K55C	119392	Cody, William F.
Hernandez, Mauricio A.	0A8359	22/K55C	043331	Haas, Laura M.

Inventors without Lotus Notes IDs

Renee J Miller
Serial Number : (N/A) Company : University of Toronto
Citizen of : US
E-Mail : miller@cs.toronto.edu
Business Address :
Department of Computer Science
University of Toronto
10 King's College Road
Toronto, Ontario
CANADA M5S 3G4
Business Phone : (416)946-3621
Home Address :

IDT Selection

IDT Team:	Attorney/Patent Professional:
Marc D McSwain/Almaden/IBM Cheryl Ruby/Almaden/IBM	Marc D McSwain/Almaden/IBM

Response Due to IP&L : 06/14/2000

Main Idea

***Title of disclosure (in English)**
Schema Mapping as Query Discovery

***Idea of disclosure**
1. Describe your invention, stating the problem solved (if appropriate), and indicating the advantages of using the invention.
Modern data intensive applications including data warehousing, global information systems and electronic commerce, require the interoperation of several heterogeneous components, each having its own individual representation of data. To enable these applications to deal with this heterogeneity, we must solve the *schema mapping* problem in which a source (legacy) data representation is mapped into a different, but fixed, target schema. Schema mapping involves the discovery of a query or set of queries that transform the source data into the new structure. We introduce an interactive mapping creation paradigm that relies on the use of *value correspondences* that show how a value of a target attribute can be created from a set of values of source attributes. We have implemented this mapping creation paradigm in *Clio*, a prototype tool for semi-automated schema mapping. **This disclosure claims the *incremental algorithm* for schema mapping at the heart of Clio as a new invention.**

Two clear advantages of using this algorithm for schema mapping are:

1. Ease of use: The user works with relationships between individual source and target attribute values and lets *Clio* generate the possibly complex SQL statements that realize the mapping.
2. Generality & Power: The algorithm handles complex mappings involving joins, aggregates, nesting, and set-theoretic operations like union, intersection, and difference.

2. How does the invention solve the problem or achieve an advantage,(a description of "the invention", including figures inline as appropriate)?

paper.pd

We present an algorithm that guides a user through the process of mapping a source schema into a target schema. The details of the algorithm can be found in Section 4 of the attached paper. We summarize the algorithm here.

We claim the incremental algorithm at the core of Clio as a new mechanism to guide users through the process of schema mapping. This algorithm takes as input a set of *value correspondences* and produces as output a *viewdefinition* that expresses the target schema as a function of the source schema. A value correspondence is a function defining how a value (or combination of values) from a source database can be combined to form a value in the target. For example, a string concatenation function can be used to indicate that a value of a staff-id attribute of a target schema is formed by concatenating the letter 'E' to an employee number from the source. Value correspondences may be entered by the user or may be suggested by Clio through some discovery process. Because value correspondences pertain to a single target attribute, they are simple for users to understand and construct, as opposed to manually constructing SQL views.

Given a target relation and a set of value correspondences that define how the values of that target relation are constructed from values in some source relations, the algorithm performs the following steps:

1. The value correspondences are divided into *potential candidate sets*. Each potential candidate set represents a single way of mapping the attributes in a target relation. By definition there is at most one value correspondence per attribute of the target relation in a potential candidate set.
2. Potential candidate sets are examined to see if a join condition is needed. If the value correspondences in a set map source values from several source relations, a join condition (a connection) between the source relations will need to be discovered. Potential candidate sets for which a join condition can be found are now called **candidate sets**.
3. Candidate sets are ranked and combined into **covers** for the target relation. A cover is a subset of candidate sets such that every value correspondence that maps an attribute of the target relation appears *at least once* in that subset. If multiple covers exists, they are ranked and presented to the user for evaluation.
4. The selected cover is converted to a query view definition (currently, a SQL view) that can be used to populate the target relation.

An incremental version of the algorithm (which is what is implemented in *Clio*) uses the same steps. This interactive and incremental algorithm takes as input the currently selected cover and a single modification to the set of value correspondences (e.g., a new value correspondence or the deletion of an existing value correspondence). The result of a single iteration of the incremental algorithm is a new cover that includes the incremental modification. A set of heuristics, described in the attached paper, guide the search for new covers in what is, in the worst case, an exponential search.

We claim the following inventions:

1. An interactive algorithm that guides the user towards the *most likely* schema mapping using simple value correspondences.
2. A division of the process into four interactive steps.
3. A set of heuristics that guide ranking the results of each step. This ranking of results helps guide the search for the most likely mapping inside an exponentially large set of possible mappings.
4. An incremental version of the algorithm.
5. A schema mapping tool that can handle value correspondences taking as input values from multiple sources and with value mappings that provide different definitions for the same target value.

3. If the same advantage or problem has been identified by others (Inside/outside IBM), how have those others solved it and does your solution differ and why is it better?

Related work falls into two major classes: schema integration and data transformation tools. Schema integration research focuses on the problem of converging two or more distinct source schemas and does not typically worry about creating mappings from the source schemas to the converged schema (see attached paper for references). Few commercial schema integration tools exists. Evoke's *Migration ArchitectTM* (www.evokesoft.com) is one. *Migration ArchitectTM* automatically collects dependency information from legacy data sources and guides the user towards the construction of a normalized target (relational) schema. Mappings between source and target schemas are tracked as the normalized target schema is created. Because the target schema is created from the source, mappings are typically quite simple.

Data transformation tools allow users to specify correspondences between a source and a target schema. Typically, these tools generate programs (code) to capture the needed transformations and, either restrict the user to correspondences between a singe source and a single target, or, if multiple sources are